



TITLE:

有理型微分方程式の変換(数式処理と数学研究への応用)

AUTHOR(S):

大河内, 茂美

CITATION:

大河内, 茂美. 有理型微分方程式の変換(数式処理と数学研究への応用).
数理解析研究所講究録 1992, 811: 127-133

ISSUE DATE:

1992-10

URL:

<http://hdl.handle.net/2433/83025>

RIGHT:

有理型微分方程式の変換

大分大工 大河内茂美 (Shigemi Ohkohchi)

ここでは、微分方程式系として

$$X' = A(z)X$$

を考えることにする。独立変数 z は複素変数としておく。
また、係数行列 $A(z)$ は $z = \infty$ で有理型関数からなる行列としておく。つまり、 $z = \infty$ ではその特異性は高々極であり、そのまわりでは、正則な関数であるとする。このような微分方程式を有理型微分方程式と呼ぶことにして、その形式的に標準的であろうと思われる姿を考えることにする。また、ここでは、行列のサイズはとりあえずは 2×2 の行列であるとしておく。

つぎに、有理型微分方程式を変換する行列として、 $T(z)$ を考える。 $T(z)$ はその逆行列 $T^{-1}(z)$ とともに、 $z = \infty$ で有理型な行列であるとしておく。このような変換行列を有理型変換とよぶことにする。このとき、有理型変換を用いて作った $Y(z) = T^{-1}(z)X(z)$ は微分方程式

$$Y'(z) = B(z)Y(z)$$

ただし $B(z) = T^{-1}(z)A(z)T(z) - T^{-1}(z)T'(z)$

の解であるから、有理型変換によって、有理型微分方程式はまたそこに変換されていることが分かる。従って、有理型微分方程式を有理型変換によって分類することにより、標準型を導き出すができる。

このような見方での、基本的な出発点は、バーコフの結果であって、それは、標準型である $B(z)$ ととして次のような形をもつとしてよいというものである。 ” $B(z)$ は $C - \{0\}$ で正則であり、 $z=0$ で第一種の特異点をもつ ” ここで、第一種の特異点というのは、 $B(z) = (z-a)^{-p-1}H(z)$ とするとき、 $H(z)$ は a で正則であり、 $H(a) \neq 0$ として、 $p=0$ ならば、第一種の特異点、 $p \geq 1$ ならば、第二種の特異点と呼ばれる意味で用いている。

これで、有理型微分方程式に対する標準型の大きな枠は設定された訳であるが、さらに厳しく型を定めるために W.B. Jurkat は NSE(Normalized Standard Equation) なるものを提案した。それは、係数行列が以下の条件を満足するというものである。

- ① $A(z) = \sum_{k=-1}^{r-1} A_k z^k$
- ② $\text{tr } A(z) = c/z \quad c \in C$
- ③ A_{r-1} は異なる固有値をもつ。

- ④ A_{-1} の固有値には整数差がないかもしくは等しい。
- ⑤ A_{-1} がスカラー行列ならば、 A_{r-1} は対角行列である。
- ⑥ A_{-1} および A_{r-1} は互いに反対向きの三角行列である。
- ⑦ $A(z)$ の成分である $a_{12}(z)$ は monic である。ただし、
 $a_{12}(z)=0$ の場合は $a_{21}(z)$ が monic である。

このような条件で定まる NSE のなかでの有理型変換によって、互いに移りあう関係を調べることによって、標準型をより詳しく求めることができるであろう。たとえば、

$$\text{diag } A_{r-1} = \text{diag}(\lambda, \lambda), \text{diag } A_{-1} = \text{diag}(\mu_1, \mu_2)$$

としたとき、 $T(z)$ として定数変換で、 μ_1, μ_2 の順序は交換して、 A_{r-1} はそのままに保つ変換等を容易に構成できることがわかる。

また、有理型微分方程式 $X'(z) = A(z)X(z)$ の原点での基本解は $X(z) = G(z)z^M$ の形であって、 M は A_{-1} のジョルダン標準型であり、 $G(z)$ は正則で、 $\det G(0) \neq 0$ である。無限遠点での形式解は $X(z) = F(z) z^\Lambda \exp[Q(z)]$ である。ここで、

$$Q(z) = \text{diag}(q(z), -q(z))$$

$$q(z) = (\lambda/r)z^r + \dots$$

$$\Lambda = \text{diag}(\lambda_1, \lambda_2) \quad \lambda_1 + \lambda_2 = \mu_1 + \mu_2$$

$$F(z) = \{f_{ij}(z)\} \quad z^{-1} \text{ のべき級数}$$

である。これらのことから、有理型変換として多項式によ

る変換を実行すれば、 $\lambda_1, \lambda_2, \mu_1, \mu_2$ を整数分だけ変化させることがわかる。さらに、つぎの事実が成立する。

定理 有理型微分方程式 $X'(z)=A(z)X(z)$ と正の整数 k に対して、多項式 $T(z)$ ($\det T(z)$ は一定) で、方程式を $Y'(z)=B(z)Y(z)$ に変換し、 $\Lambda_A - \Lambda_B = \text{diag}(k, -k)$ が存在する。

\Leftrightarrow つぎの条件を満足する多項式 $p(z), q(z)$ が存在することである。

$\deg p(z)=k, \deg q(z)<k$ で、 $p(z)$ は monic

$$q(z)f_{11}(z)+p(z)f_{21}(z)=0(z^{-k-1})$$

この事実を用いて、有理型変換として多項式変換を用いた場合の同値な有理型微分方程式を定めることが出来る。実際、この定理での多項式 $p(z), q(z)$ と $r(z), s(z)$ を用いて

$$T(z) = \begin{pmatrix} p(z) & -r(z) \\ -q(z) & s(z) \end{pmatrix}$$

$$p(z)s(z) - q(z)r(z) = 1$$

$$\deg(r) < \deg(p) \quad \deg(s) < \deg(q)$$

として計算すればいい。整数 k に関しては、 $f_{21}(z)$ の次数で考えれば良いのであるが、 $\deg f_{21}(z) = \deg a_{21}(z) - (r-1)$

) を利用すれば簡単にわかる。

具体例として

$$A(z) = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} z + \begin{pmatrix} m_1 & m_2 \\ 0 & -m_1 \end{pmatrix} + \begin{pmatrix} n_1 & n_2 \\ n_3 & n_4 \end{pmatrix} z^{-1}$$

で係数行列が与えられる場合には、 $k=2$ とする多項式変換で

$$B(z) = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} z + \begin{pmatrix} m_1 & & 0 \\ & -n_3 n_3 m_2 / 4 & -m_1 n_3 / 2 \\ & & -m_1 \end{pmatrix} + \begin{pmatrix} n_1 - 2 & & -4/n_3 \\ & n_3(-2n_1 - n_2 n_3 + 2n_4 + 4)/4 & n_4 + 2 \end{pmatrix} z^{-1}$$

と変換されるので、 $n_3 m_2 + 2m_1 n_3 = 0$ 、 $-2n_1 - n_2 n_3 + 2n_4 + 4 = 0$ の場合には求積可能であることもわかる。この結果を単独の方程式

$$z^2 y'' = z(2z^2 + bz + d)y' + (z^2 + az + c)y$$

に適用してみた場合には、 $a=0, d-2c+5/2=0$ は求積可能であることになる。さらに、変換後の $B(z)$ に対して、定数変換と独立変数の変換を施せば $A(z)$ と同じ形になるから、同様の計算で同値な有理型方程式を次々と求めていくことが出来る。また、REDUCEでの出力例を以下にしめておく。

```

%%%%% EXAMPLE %%%%%%%%%%%%%%
a11:=z**2+m1*z+n1;

```

```

A11 := M1*Z + N1 + Z2
a22:=-z**2-m1*z+n4;

```

```

A22 := - M1*Z + N4 - Z2
a12:=m2*z+n2;
A12 := M2*Z + N2
a21:=n3;
A21 := N3

```

```

%%%%% EXAMPLE %%%%%%%%%%%%%%

```

```

p:=z**2+p3*z+p4;

```

```

P := P3*Z + P4 + Z2

```

```

q:=q3*z+q4;

```

```

Q := Q3*Z + Q4

```

```

r:=r3*z+r4;

```

```

R := R3*Z + R4

```

```

s:=s4;

```

```

S := S4

```

```

b11:=s*a11*p-s*a12*q-s*df(p,z)*z+r*a21*p-r*q*a22+r*df(q,z)*z$

```

```

b12:=-s*a11*r+a12*s*s+s*df(r,z)*z-r*r*a21+s*r*a22-r*df(s,z)*z$

```

```

b21:=a11*p*q-a12*q*q-q*df(p,z)*z+a21*p*p-p*q*a22+p*df(q,z)*z$

```

```

b22:=-a11*q*r+a12*q*s+z*q*df(r,z)-p*r*a21+p*s*a22-p*z*df(s,z)$

```

```

bb:=p*s-r*q-1$

```

```

q3:=0;

```

```

Q3 := 0

```

```

s4:=0;

```

```

S4 := 0

```

```

r3:=0;

```

```

R3 := 0

```

```

q4:=-n3/2;

```

```

Q4 := ( - N3)/2

```

```

p3:=m1;

```

```

P3 := M1

```

```

r4:=2/n3;

```

```

R4 := 2/N3

```

```

p4:=-1+(n1-n4)/2;

```

```

P4 := (N1 - N4 - 2)/2

```

```

array ma(10);

```

```

  for i:=0 : 10 do

```

```

    begin

```

```

      ma(i):=sub(z=0,df(b11,z,i));

```

```

    end;

```

```

array tu(10);

```

```

  for i:=0 : 10 do

```

```

    begin

```

```

    tu(i):=sub(z=0,df(b12,z,i));
end;

array sa(10);
for i:=0 : 10 do
begin
    sa(i):=sub(z=0,df(b21,z,i));
end;

array ka(10);
for i:=0 : 10 do
begin
    ka(i):=sub(z=0,df(bb,z,i));
end;

for i:=0:10 do write ma(i);
N1 - 2
M1
2
0

for i:=0:10 do write tu(i);
( - 4)/N3
0
0

for i:=0:10 do write sa(i);
(N3*( - 2*N1 - N2*N3 + 2*N4 + 4))/4
(N3*( - 2*M1 - M2*N3))/4
0

for i:=0:10 do write ka(i);
0
b11;
M1*Z + N1 + Z2 - 2
b12;
( - 4)/N3
b21;
(N3*( - 2*M1*Z - M2*N3*Z - 2*N1 - N2*N3 + 2*N4 + 4))/4
b22;
- M1*Z + N4 - Z2 + 2
p;
(2*M1*Z + N1 - N4 + 2*Z2 - 2)/2
q;
( - N3)/2
r;
2/N3
s;
0
end;

```